designated as being protected. The protection occurs in response to encode icon 98 being activated with a click of a mouse. For example, display 68 and 74 are enclosed in a rectangle indicating that segments 2 and 5 (which include images I2 and I5) will be protected if encode icon 98 is activated. Activation of a watermark icon 100 causes information such as is described below to be contained in a watermark.

Amend the paragraph at page 8, lines 12-17, as follows:

FIG. 4 illustrates a content providing system 114 which is similar to content providing system 14 but illustrates some additional capabilities, which could be included in content providing system 14. A segment creation mechanism 120 represents a user interface and associated software to select segments of the group of segments (e.g., to designate the beginning and ending frames or time of the segment). Mechanism 120 may be used for joining disjointed segments in a group and/or dividing continuous content into segments of a group.

Amend the paragraph at page 15, line 19, to page 16, line 2, as follows:

FIG. 11 illustrates a computer 220 (which may be an example of system 14) including a processor 222, on-die memory 224, chipset I/O 227, and off-die memory 228. Memory 224, memory 228, and a disc 230 include machine readable media to hold instructions to be executed and other data. The various block diagram and flow chart blocks in the other figures called mechanisms may represent processor 222 performing functions on software or may represent hardware other than processor 222 performing the functions described in connection with the block diagram or flowchart mechanisms. A link 234 joins computer 220 to a remote computer 236 (which may be an example of remote receiving computer 20). Computer 236 may be the same as of different than computer 220. A display 238 may be packaged with or separate from computer 236. Link 234 represents any of various links including the Internet, an intranet, a local area network, satellite, or other networks. The term computer is intended to be broadly interpreted to include a variety of systems and devices including personal computers, mainframe computers, set top boxes, digital versatile disc (DVD) players, and the like.

Amend the paragraph at page 20, lines 14-18, as follows:

The invention may be used with respect to signals not previously compressed. FIG. 15 illustrates an encode mechanism 270 in which uncompressed (raw) video is first transformed with a DCT mechanism 272 (which may be the same as encoder 200 in FIG. 9). Scrambling mechanism 244 alters the coefficients as described above. An inverse DCT mechanism 276 returns the scrambled video to the uncompressed (raw) video format. Selected coefficients are provided by coefficient selection mechanism 264 responsive to a key and strength parameter 266.

Amend the paragraph at page 21, lines 12-21, as follows:

As an example, FIG. 17 illustrates a scrambling encode mechanism 300 (which may be in computer 220 in FIG. 11) in which video blocks (which may be in MPEG format) are received by in buffer 302. In some embodiments, as a block is received, it is identified with a number m or placed in position m of the buffer. The number m is incremented by increment mechanism 308 with each received block until m = N (compare mechanism 306), where N is the number of blocks available for permutation. For example, if a set of four blocks may be permuted, N is 3 (assuming m starts at 0). When m = N, order selection mechanism 312 selects a block order based on a key and sets m to 0 (mechanism 316). The blocks are read from buffer 302 in the permuted block order as specified in the block order from order selection mechanism 312. The block order may be a mapping for each block, wherein or not it is changed or only those that change order.

Amend the paragraph at page 21, lines 22-28, as follows:

FIG. 18 illustrates a descrambling decode mechanism 320 (which may be in computer 236 in FIG. 11) which receives the blocks in permuted order in buffer 322 from buffer 302 in FIG. 17. When the buffer is full (comparison mechanism 326), order selection mechanism 332 selects the block order responsive to a key and buffer 322. Responsive to the block order, the blocks in the original order are read from buffer 322 in the original order. Mechanism 328 increments m and mechanism 336 sets m = 0. By using the same block order as in FIG. 17, an inverse permutation occurs and the blocks are read out in the original order.